МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Тульский государственный университет»

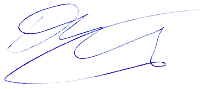
Институт *прикладной математики и компьютерных наук*

Кафедра *вычислительной техники*

Курсовая работа по дисциплине

Численные методы

на тему: Численные методы решения дифференциальных уравнений

Студент группы 220681 Шевердин Д.О 22.06.20

(Ф.И.О.) (Подпись, дата)

Руководитель работы Волошко А. Г. \_\_ \_\_\_\_\_\_\_\_\_\_\_

(Ф.И.О., должность) (Подпись, дата)

Комиссия: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тула 2020

МИНОБРНАУКИ РОССИИ

ФГБОУ ВО «Тульский государственный университет»

Институт прикладной математики и компьютерных наук

Кафедра «Вычислительная техника»

**З А Д А Н И Е**

На курсовой проект (курсовую работу) по дисциплине (наименование дисциплины указать полностью) \_Численные методы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

студенту(ке) группы \_220681\_\_\_\_\_ Ф.И.О. студента \_Шевердин Даниил Олегович\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тема проекта (работы) \_Численное решение дифференциальных уравнений\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

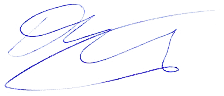
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Входные данные \_Методы Адамса и Рунге-Кутта. Уравнение . \_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Задание получил  «\_\_15\_\_» \_\_февраля\_\_\_\_\_\_\_\_\_20\_20\_г.

(подпись студента)

Срок предоставления задания «\_\_23\_\_» \_\_\_\_\_\_\_\_июня\_\_\_\_\_\_\_\_\_\_\_\_ 20\_20\_г.

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись) (расшифровка подписи)

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_г.

К защите. Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись) (расшифровка подписи)

Замечания руководителя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_г.

*При защите курсового проекта (работы) наличие рецензии обязательно.*

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc43727315)

[1. ЗАДАНИЕ 5](#_Toc43727316)

[2. МАТЕМАТИЧЕСКОЕ ОПИСАНИЕ МЕТОДОВ 5](#_Toc43727317)

[3. ОПИСАНИЕ ВХОДНЫХ И ВЫХОДНЫХ ДАННЫХ 10](#_Toc43727318)

[4. АЛГОРИТМЫ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ 10](#_Toc43727319)

[5. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ 12](#_Toc43727320)

[6. РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ 16](#_Toc43727321)

[7. ПРОВЕРКА КОРРЕКТНОСТИ РАБОТЫ ПРОГРАММЫ 18](#_Toc43727322)

[8. СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ПО КРИТЕРИЯМ ТОЧНОСТИ, ВЫЧИСЛИТЕЛЬНОЙ СЛОЖНОСТИ 20](#_Toc43727323)

[ЗАКЛЮЧЕНИЕ 27](#_Toc43727324)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 28](#_Toc43727325)

[ПРИЛОЖЕНИЕ 29](#_Toc43727326)

[MyTable.cs 29](#_Toc43727327)

[MainWindow.xaml.cs 29](#_Toc43727328)

# ВВЕДЕНИЕ

Целями и задачами работы является получения навыков:

* анализа различных численных методов решения дифференциальных уравнений;
* разработки программных средств для решения численными методами дифференциальных уравнений.

В ходе выполнения курсовой работы необходимо разработать программу для вычисления значения дифференциального уравнения и погрешности методами Адамса и Рунге-Кутта.

# 1. ЗАДАНИЕ

Разработать на языке программирования C# программу, реализующую решение уравнения на отрезке [a, b] ([a, b] = [x0, xn]) с шагом h методом Рунге-Кутта и методом Адамса. Начальные значения x и y, конечное значение xn, величина шага h задаются пользователем.

# 2. МАТЕМАТИЧЕСКОЕ ОПИСАНИЕ МЕТОДОВ

МЕТОД РУНГЕ-КУТТА

Метод Рунге-Кутта является одним из методов повышенной точности.

Пусть на отрезке [а, b] требуется найти численное решение уравнения

y' = f (x, y) (1)

с начальным условием

y(x0) = y0 (2)

Разобьём отрезок [а, b] на n равных частей точками xi = x0 + ih (i = 1,2,..., n, a h = (b - a) / n - шаг интегрирования). В методе Рунге-Кутта последовательные значения у искомой функции у определяются по формуле:

yi+1 = yi + ∆yi (3)

Если разложить функцию у в ряд Тейлора и ограничиться членами до h4 включительно, то приращение функции ∆y можно представить в виде

∆𝑦=𝑦(𝑥+ℎ)−𝑦(𝑥)=ℎ𝑦′(𝑥)+ℎ2/2\*𝑦′′(𝑥)+ℎ3/6\*𝑦′′′(𝑥)+ℎ4/24\*𝑦′′′′(𝑥) (4)

где производные y''(x), y'''(x), y''''(x) находят последовательным дифференцированием из уравнения (3).

Вместо непосредственных вычислений по формуле (3) методом Рунге-Кутта определяют четыре числа:

𝑘1=ℎf (𝑥,𝑦); 𝑘2=ℎf(𝑥+ ℎ/2,𝑦+𝑘1/2); 𝑘3=ℎf (𝑥+ ℎ/2,𝑦+𝑘2/2); 𝑘4=ℎf (𝑥+ℎ,𝑦+𝑘3) (5)

Можно доказать, что если числам k1, k2, k3, k4 придать соответственно веса 1/6; 1/3; 1/3; 1/6, то средневзвешенное этих чисел, т. е.

(1/6)\*𝑘1+(1/3)\*𝑘2+(1/3)\*𝑘3+(1/6)\*𝑘4 (6)

с точностью до четвёртых степеней равно значению ∆y , вычисленному по формуле (4):

∆y=(1/6)\*(𝑘1+2𝑘2+2𝑘3+𝑘4) (7)

Таким образом, для каждой пары текущих значений xi и yi определяют значения

𝑘1(𝑖)=ℎf (𝑥i,𝑦i); 𝑘2(𝑖)=ℎf (𝑥i+ ℎ/2,𝑦i+𝑘1(𝑖)/2); 𝑘3(𝑖)=ℎf (𝑥i+ ℎ/2,𝑦i+𝑘2(𝑖)/2); 𝑘4(𝑖)=ℎf (𝑥i+ℎ,𝑦i+𝑘3(𝑖)) (8)

Метод Рунге-Кутта имеет порядок точности h4 на всем отрезке [а, b]. Оценка точности этого метода очень затруднительна. Грубую оценку погрешности можно получить с помощью "двойного просчёта" по формуле

|𝑦∗𝑖−𝑦(𝑥𝑖)|≈(𝑦∗𝑖−𝑦𝑖)/15 (9)

Где 𝑦(𝑥𝑖) - значение точного решения уравнения (1) в точке 𝑥𝑖 а 𝑦∗𝑖 и 𝑦𝑖 - приближенные значения, полученные с шагом h/2 и h.

Если e - заданная точность решения, то число n (число делений) для (b-a)/n определения шага интегрирования h = выбирается таким образом, чтобы

ℎ4 < e (10)

Однако шаг расчёта можно менять при переходе от одной точки к другой.

Для оценки правильности выбора шага h используют равенство:

𝑞=|(𝑘2(𝑖)−𝑘3(𝑖))/(𝑘1(𝑖)−𝑘2(𝑖))| (11)

где q должно быть равно нескольким сотым, в противном случае шаг h уменьшают.

Метод Рунге-Кутта может быть применён и к решению систем дифференциальных уравнений.

Пусть задана система дифференциальных уравнений первого порядка

{𝑦′=𝑓(𝑥,𝑦,𝑧);𝑧′=𝑔(𝑥,𝑦,𝑧) (12)

с начальными условиями

x =𝑥0 , y(𝑥0) = 𝑦0 , z(𝑥0) = 𝑧0 (13)

В этом случае параллельно определяются числа ∆yi и ∆zi:

∆𝑦𝑖=(1/6)\*(𝑘1(𝑖)+2𝑘2(𝑖)+2𝑘3(𝑖)+𝑘4(𝑖)); ∆𝑧𝑖=(1/6)\*(𝑙1(𝑖)+2𝑙2(𝑖)+2𝑙3(𝑖)+𝑙4(𝑖))} (14)

где 𝑘1(𝑖)= h𝑓(𝑥𝑖 , 𝑦𝑖 , 𝑧𝑖);

𝑙1(𝑖)= hg(𝑥𝑖 ,𝑦𝑖 ,𝑧𝑖);

𝑘2(𝑖)= hf(𝑥𝑖+ℎ/2,𝑦𝑖+𝑘1(𝑖)/2,𝑧𝑖+𝑙1(𝑖)/2);

𝑙2(𝑖)= hg(𝑥𝑖+ℎ/2,𝑦𝑖+𝑘1(𝑖)/2,𝑧𝑖+𝑙1(𝑖)/2);

𝑘3(𝑖)= hf(𝑥𝑖+ℎ/2,𝑦𝑖+𝑘2(𝑖)/2,𝑧𝑖+𝑙2(𝑖)/2);

𝑙3(𝑖)= hg(𝑥𝑖+ℎ/2,𝑦𝑖+𝑘2(𝑖)/2,𝑧𝑖+𝑙2(𝑖)/2);

𝑘4(𝑖)= hf(𝑥𝑖+ℎ,𝑦𝑖+𝑘3(𝑖),𝑧𝑖+𝑙3(𝑖));

𝑙4(𝑖)= hg(𝑥𝑖+ℎ,𝑦𝑖+𝑘3(𝑖),𝑧𝑖+𝑙3(𝑖)).

Тогда получим решение системы

𝑦𝑖+1=𝑦𝑖+∆𝑦𝑖, 𝑧𝑖+1=𝑧𝑖+∆𝑧𝑖 (15)

МЕТОД АДАМСА

При решении дифференциального уравнения методом Рунге-Кутта необходимо производить много вычислений для нахождения каждого. В том случае, когда правая часть уравнения сложное аналитическое выражение, решение такого уравнения методом Рунге-Кутта вызывает большие трудности. Поэтому на практике применяется метод Адамса, который не требует многократного подсчета правой части уравнения.

Пусть дано дифференциальное уравнение

, (16)

с начальным условием

, . (17)

Требуемся найти решение этого уравнения на отрезке [a.b].

Разобьем отрезок [a,b] на n равных частей точками    
(*i = 1, 2,..., n*), a  – проинтегрируем дифференциальное уравнение). Выберем участок  и проинтегрируем дифференциальное уравнение (16); тогда получим

,

или

. (18)

Для нахождения производной воспользуемся второй интерполяцион­ной формулой Ньютона (ограничиваясь при этом разностями третьего по­рядка):

. (19)

или

. (20)

Подставляя выражение для  из формулы (20) в соотношение (18) и учитывая, что , имеем

 (21)

Обозначим в дальнейшем  (*i = 0,1,2,…,n*).

Тогда для любой разности имеем  и

. (22)

По формуле  получаем решение уравнения. Формула (22) носит название *экстраполяционной формулы Адамса*.

Для начала процесса нужны четыре начальных значения  - так называемый *начальный отрезок*, который может быть найден, исходя из начального условия (17) с использованием одного из известных методов. Обычно начальный отрезок решения находится методом Рунге-Кутта.

Зная  можно определить

; ;

; . (23)

Далее составляется таблица разностей величины *q* (табл. 1).

Таблица 1 – Таблица разностей величины *q*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| I |  |  |  |  |  |  |  |  |
| (1) | (2) | (3) | (4) | (5) | (6) | (7) | (8) | (9) |
| 0 |  |  | - |  |  |  |  |  |
| 1 |  |  | - |  |  |  |  | - |
| 2 |  |  | - |  |  |  | - | - |
| 3 |  |  |  |  |  | - | - | - |
| 4 |  |  | - | - | - | - | - | - |
| 5 |  | - | - | - | - | - | - | - |
| 6 |  | - | - | - | - | - | - | - |

Метод Адамса заключается в продолжении диагональной таблицы разностей с помощью формулы (22). Используя числа , которые располагаются в таблице по диагонали, полагая в формуле (22)  
 *n = 3* (известное последнее значение *у* есть ), получаем:

.

Полученное значение  вносят и таблицу и находят . Затем используя значения  и  находят  т.е. получается новая диагональ. По этим данным вычисляют

 .

Таким образом, продолжают таблицу решения, вычисляя правую часть дифференциального уравнения (16) на каждом этапе только один раз.

Для грубой оценки погрешности применяют **принцип Рунге**, который состоит в следующем:

1. Находят решение дифференциального уравнения при шаге *h*.
2. Значение шага удваивают и находят решение при шаге *Н = 2h*.

3. Вычисляют погрешность метода по формуле

, (24)

где  - значение приближенного вычисления при двойном шаге *H=2h*;  -значение приближенного вычисления при шаге *h*.

# 3. ОПИСАНИЕ ВХОДНЫХ И ВЫХОДНЫХ ДАННЫХ

Входные данные:

1. Начальное значение для x — ассоциированная переменная x0. Тип данных — double.
2. Начальное значение для y — ассоциированная переменная y0. Тип данных — double.
3. Последнее значение для x — ассоциированная переменная xn. Тип данных - double.

Условия для входных данных:

1. Начальное значение (переменная x0) не может быть больше Последнего (переменная xn).

# 4. АЛГОРИТМЫ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

4.1. После получения входных данных в цикле производятся вычисления \_y с половиной шага half\_h = h/2. Для этого считаются k1, k2, k3 и k4 как half\_h \* f(dxi, d\_yi). При подсчёте k1 значение dxi = xi, d\_yi = yi. При подсчёте k2 значение dxi = xi+\_h/2, d\_yi = yi+k1/2. При подсчёте k3 значение dxi = xi+\_h/2, d\_yi = yi+k2/2. При подсчёте k4 значение dxi = xi+\_h, d\_yi = yi+k3. Значение \_yi считается как сумма k1, 2\*k2, 2\*k3 и k4, делённая на 6.

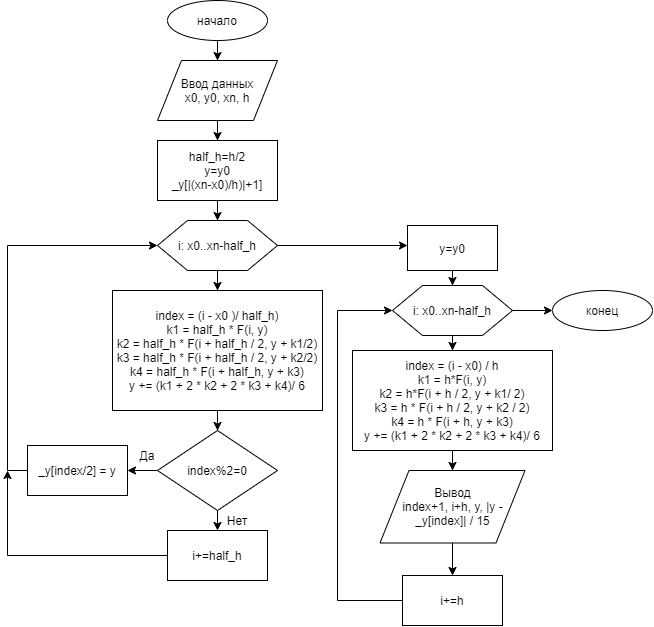
После выхода из цикла, выполняется другой цикл, в котором производятся вычисления y. Для этого считаются k1, k2, k3 и k4 как h \* f(dxi, dyi). При подсчёте k1 значение dxi = xi, dyi = yi. При подсчёте k2 значение dxi = xi+h/2, dyi = yi+k1/2. При подсчёте k3 значение dxi = xi+h/2, dyi = yi+k2/2. При подсчёте k4 значение dxi = xi+h, dyi = yi+k3. Значение yi считается как сумма k1, 2\*k2, 2\*k3 и k4, делённая на 6. Далее производится вывод полученных x, y и погрешности = |yi - \_yi|/15.

Рисунок 1 – Блок-схема алгоритма Рунге-Кутта

4.2 После получения входных данных производится поиск значений для «начального отрезка» с помощью метода Рунге-Кутта, описанного выше, но для двойного шага double\_h=2\*h. Каждое значение y начального отрезка сохраняется в массив start\_segment. После выхода из цикла рассчитывается q0, q1, q2, q3. При подсчете qi h=double\_h, f(xi,yi)=f(x0+i\*double\_h, start\_segment[i]). Производится расчет q0, q1,q2,q0, q1 и q0.

В цикле с шагом double\_h производится вычисления y по формуле Адамса, начиная с y4. Значения сохраняются в массив H для дальнейшего подсчета погрешности. Затем происходит сдвиг начального отрезка: y0 = y1, y1 = y2, y2 = y3. Аналогично происходит сдвиг qi. При подсчете q3 f(xi,yi)=f(x3, start\_segment[3]). Затем происходит расчет q0, q1,q2,q0, q1 и q0 для новых значений qi.

После описанный алгоритм повторяется для шага h. В последнем цикле выполняется подсчет погрешности, результаты выводятся в таблицу.

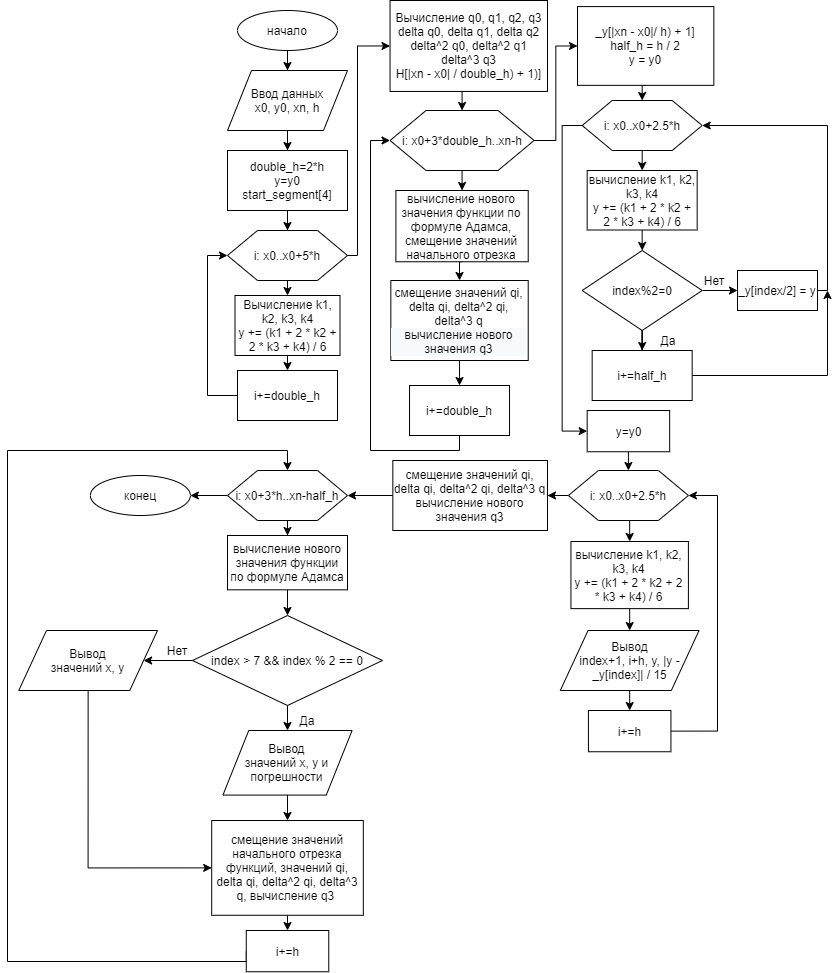


Рисунок 2 – Блок-схема алгоритма Адамса

# 5. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Основной класс содержит следующие глобальные переменные:

* double h – шаг;
* double x0, y0, xn – значения для трёх точек, вводимые пользователем.

Метод Button\_Click вызывается по нажатию кнопки «Решить уравнение». Данный метод принимает введенные пользователем переменные h, x0, y0, xn, проверяет корректность данных и вызывает метод для расчета в зависимости от выбранного элемента checkBox.

Метод checkBox1\_Checked срабатывает при выборе метода. Он снимает флажок с другого метода. Аналогичным образом работает метод checkBox2\_Checked. Метод F задаёт функцию.

Метод Runge вычисляет значения для x, y и погрешность, заносит их в таблицу. Также метод вычисляет затраченное на вычисление время и заносит его в объект label.

Метод Adams вычисляет значения для x, y и погрешность и заносит их в таблицу. Также метод вычисляет затраченное на вычисление методом Адамса время и заносит его в объект label.

# 6. РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ

При запуске программы откроется окно, в котором необходимо выбрать нужный метод, ввести начальные значения и нажать кнопку «Решить уравнение». При подсчёте методом Рунге-Кутты программа продемонстрирует следующие результаты (Рисунок 3).

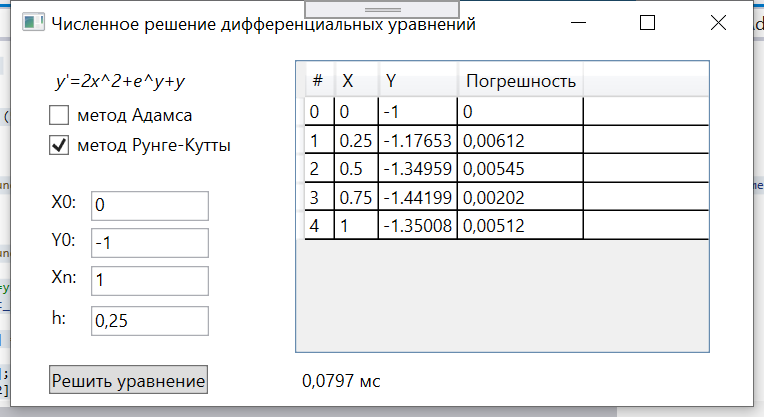


Рисунок 3 – Результаты при подсчёте методом Рунге-Кутты

При подсчёте методом Адамса программа продемонстрирует следующие результаты (Рисунок 4).

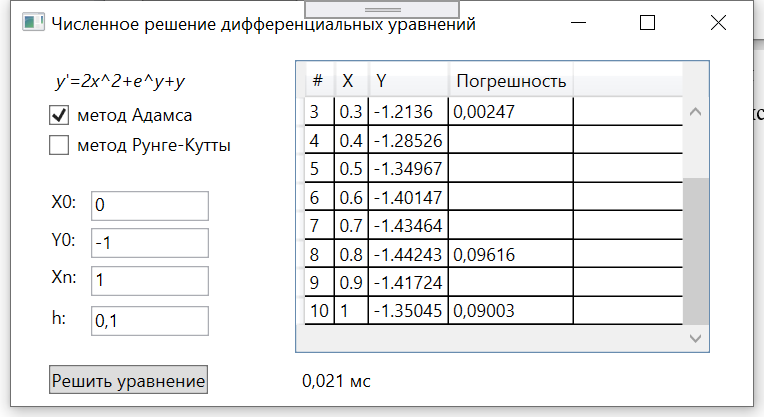


Рисунок 4 – Результаты при подсчёте методом Адамса

# 7. ПРОВЕРКА КОРРЕКТНОСТИ РАБОТЫ ПРОГРАММЫ

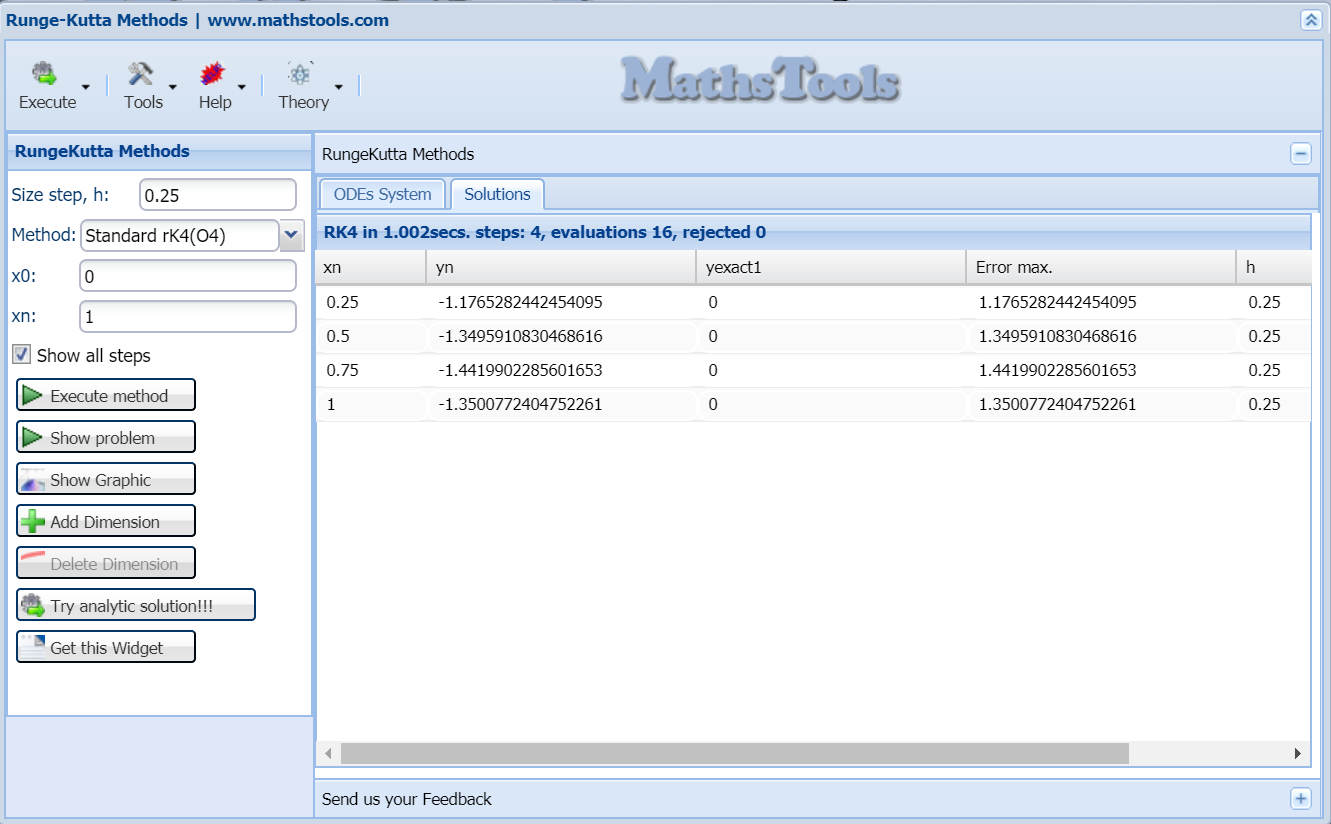
7.1 Проверка корректности работы программы для метода Рунге-Кутты с помощью онлайн калькулятора MathsTools с сайта www.mathstools.com. Программа при таких данных дает следующий результат:

x0 = 0, x1 = 0.25, x2 = 0.5, x3 = 0.75, x4 = 1;

y0 = -1, y1 = -1.17653, y2 = -1.34959, y3 = -1.44199, y4 = -1.35008;

Погрешность0 = 0, Погрешность1 = 0.00612, Погрешность2 = 0.00545, Погрешность3 = 0.00202, Погрешность4 = 0.00512.

Результат проверки приведен на рисунке 5.

Рисунок 5 – Проверка корректности работы программы для метода Рунге-Кутты

7.2 Проверка корректности работы программы для метода Адамса с онлайн калькулятора с сайта atozmath.com. Программа при таких данных дает следующий результат:

x0 = 0, x1 = 0.1, x2 = 0.2, x3 = 0.3, x4 = 0.4, x5 = 0.5, x6 = 0.6, x7 = 0.7, x8 = 0.8, x9 = 0.9, x10 = 1;

y0 = -1, y1 = -1.06702, y2 = -1.13945, y3 = -1.2136, y4 = -1.28526, y5=-1.34967, y6 = -1.40147, y7 = -1.43464, y8 = -1.44243, y9 = -1.41724, y10 = -1.35045;

Погрешность0 = 0, Погрешность1 = 0.00612, Погрешность2 = 0.00545, Погрешность3 = 0.00202, Погрешность8 = 0,09616, Погрешность10 = 0,09003.

Результат проверки приведен на рисунке (Рисунок 6).

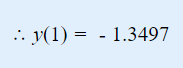


Рисунок 6 – Проверка корректности работы программы для метода Рунге-Кутта

# 8. СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ ПО КРИТЕРИЯМ ТОЧНОСТИ, ВЫЧИСЛИТЕЛЬНОЙ СЛОЖНОСТИ

Тестирование методов производилось на компьютере со следующими характеристиками:

* Операционная система Windows 10;
* Процессор Intel(R) Core(TM) i5-8250U с частотой 1.60 GHz;
* Оперативная память 8 ГБ;
* Видеокарта Intel(R) UHD Graphics 620.

Метод Рунге-Кутта, в отличие от метода Адамса является одним из методов повышенной точности, Метод Рунге-Кутта имеет порядок точности 4 на всем отрезке [а,b]. При решении дифференциального уравнения методом Рунге-Кутта необходимо производить много вычислений для нахождения каждого. В том случае, когда правая часть уравнения сложное аналитическое выражение, решение такого уравнения методом Рунге-Кутта вызывает большие трудности. Метод Рунге-Кутта может быть применен и к решению систем дифференциальных уравнений.

Практически метод Адамса применяется чаще. Он не требует многократного подсчета правой части уравнения.

Преимущество метода Адамса перед методом Рунге-Кутта заключаются в меньшей трудоемкости вычислений на один шаг.

Основные недостатки: нестандартное начало счета, невозможность (без усложнения формул) в процессе счета изменить, начиная с какой-то точки, шаг с которым ведутся вычисления. Последний пункт важен в тех случаях, когда решение и его производные на некоторых участках меняются быстро, а на других изменяются медленно.

Метод Адамса применяется также и для решения систем дифференциальных уравнений и дифференциальных уравнений n-гo порядка.

Для сравнения скорости вычислений в программу был добавлен таймер и вывод его результата для каждого из методов. Таким образом, при y0 = -1, x0 = 0, xn = 1, h = 0.1 методом Адамса решение было найдено в среднем за 0,01784 миллисекунд по результатам 5 экспериментов (табл. 2). Для этих же значений, решение методом Рунге-Кутта было найдено в среднем за 0,06614 миллисекунд (табл. 2), т. е. больше чем для метода Адамса на 0,0483 миллисекунд. При решении уравнения методом Адамса, средняя погрешность равна 0,093095 (табл. 3). В то время как для метода Рунге-Кутта средняя погрешность равна 0,001315 (табл. 3), т.е. меньше средней погрешность для метода Адамса на 0.09178. Соответственно можно сделать вывод, что метод Рунге-Кутта имеет большую точность чем метод Адамса, однако требует больше времени на вычисления.

Таблица 2 – Время для вычислений методами Адамса и Рунге-Кутты

|  |  |  |
| --- | --- | --- |
| Номер эксперимента | Метод Адамса, мс | Метод Рунге-Кутты, мс |
| 1 | 0,0074 | 0,2071 |
| 2 | 0,0105 | 0,0227 |
| 3 | 0,0213 | 0,0291 |
| 4 | 0,015 | 0,0395 |
| 5 | 0,035 | 0,0323 |

Таблица 3 – Погрешность для методов Адамса и Рунге-Кутты

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x | y (Адамса) | y (Рунге-Кутты) | Погрешность (Адамса) | Погрешность (Рунге-Кутты) |
| 0 | -1 | -1 | 0 | 0 |
| 0,1 | -1.06702 | -1.06702 | 0,00229 | 0,00229 |
| 0,2 | -1.13945 | -1.13945 | 0,00244 | 0,00244 |
| 0,3 | -1.2136 | -1.2136 | 0,00247 | 0,00247 |
| 0,4 | -1.28526 | -1.28525 | - | 0,00235 |
| 0,5 | -1.34967 | -1.34965 | - | 0,00206 |
| 0,6 | -1.40147 | -1.40142 | - | 0,0016 |
| 0,7 | -1.43464 | -1.43456 | - | 0,00092 |
| 0,8 | -1.44243 | -1.44231 | 0,09616 | 2E-05 |
| 0,9 | -1.41724 | -1.41705 | - | 0,00115 |
| 1 | -1.35045 | -1.35016 | 0,09003 | 0,00262 |

# ЗАКЛЮЧЕНИЕ

В ходе выполнения данной работы были анализированы 2 численных метода решения дифференциальных уравнений: метод Адамса и метод Рунге-Кутта. Было разработано программное средство для решения численными методами Адамса и Рунге-Кутта дифференциальных уравнений.

Также были выполнены цели и задачи работы: получены навыки анализа различных численных методов решения дифференциальных уравнений; получены навыки разработки программных средств для решения численными методами дифференциальных уравнений.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Википедия. Метод Рунге-Кутты [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/Метод\_Рунге\_—\_Кутты.

2. MathsTools [Электронный ресурс]. Режим доступа: www.mathstools.com.

3. Студопедия. Численное решение ОДУ методом Рунге-Кутты 2 порядка [Электронный ресурс]. Режим доступа: https://studopedia.ru/21\_47207\_chislennoe-reshenie-odu-metodom-runge-kutti—poryadka.html.

4. Википедия. Метод Адамса [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/Метод\_Адамса

1. Metanit [Электронный ресурс] : Руководство по WPF . Режим доступа: https://metanit.com/sharp/wpf/.
2. Шилдт Г. Язык программирования C# 4.0: полное руководство – Litres, 2019.

# ПРИЛОЖЕНИЕ

# MyTable.cs

namespace CH\_M

{

internal class MyTable

{

public MyTable(int i, double x, double y, string accuracy)

{

this.i = i;

this.x = x;

this.y = y;

this.accuracy = accuracy;

// this.F = F;

}

public double i { get; set; }

public double x { get; set; }

public double y { get; set; }

public string accuracy { get; set; }

// public double F { get; set; }

}

}

# MainWindow.xaml.cs

using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.Windows;

namespace CH\_M

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

double x0, y0, xn, h;

public MainWindow()

{

InitializeComponent();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

if (checkBox1.IsChecked == true || checkBox2.IsChecked == true)

{

try

{

x0 = Convert.ToDouble(textBox1.Text);

y0 = Convert.ToDouble(textBox2.Text);

xn = Convert.ToDouble(textBox3.Text);

h = Convert.ToDouble(textBox4.Text);

if (x0 >= xn)

{

MessageBox.Show("Первое значение х должно быть меньше последнего", "Ошибка");

return;

}

}

catch

{

MessageBox.Show("Неверный формат данных.", "Ошибка");

return;

}

if (checkBox2.IsChecked == true)

{

Runge();

}

else if (Math.Abs((xn - x0) / h) < 4)

{

MessageBox.Show("В методе Адамса первые 4 значения (начальный отрезок) находятся любым другим методом (Рунге-Кутты для этого приложения).\nЧтобы увидеть значения, вычисленные методом Адамаса, увеличьте правую границу или уменьшите шаг, так, чтобы |(Xn-Xo)|/h было больше или равно 4", "Внимание!");

Adams();

}

else

Adams();

}

else

MessageBox.Show("Выберете метод решения", "Ошибка");

}

private void Adams()

{

double double\_h = 2 \* h;

double y = y0;

double[] start\_segment = new double[4];//массив для хранения значений начально отрезка для метода Адамса

//поиск значений начального отрезка (методом рунге-кутты)

for (var i = x0; i < x0 + (5 \* h); i += double\_h)

{

int index = Convert.ToInt32((i - x0) / double\_h);

start\_segment[index] = y;//Значение y для началального отрезка

double k1 = double\_h \* F(i, y);

double k2 = h \* F(i + double\_h / 2, y + k1 / 2);

double k3 = h \* F(i + double\_h / 2, y + k2 / 2);

double k4 = h \* F(i + double\_h, y + k3);

y += (k1 + 2 \* k2 + 2 \* k3 + k4) / 6;

}

start\_segment[3] = y;//значение y3 для начального отрезка

//расчет q0, q1, q2, q3

double[] q = new double[] { double\_h \* F(x0, start\_segment[0]), double\_h \* F(x0 + double\_h, start\_segment[1]), double\_h \* F(x0 + (double\_h \* 2), start\_segment[2]), double\_h \* F(x0 + 3 \* double\_h, start\_segment[3]) };

//расчет delta q0, delta q1, delta q2

double[] d\_q = new double[] { q[1] - q[0], q[2] - q[1], q[3] - q[2] };

//расчет delta^2 q0, delta^2 q1

double[] dd\_q = new double[] { d\_q[1] - d\_q[0], d\_q[2] - d\_q[1] };

//расчет delta^3 q0

double ddd\_q = dd\_q[1] - dd\_q[0];

double[] H = new double[Convert.ToInt32(Math.Abs((xn - x0) / double\_h) + 1)];

for (var i = x0 + (double\_h \* 3); i < xn - h; i += double\_h)

{

int index = Convert.ToInt32((i + double\_h) / double\_h);

start\_segment[3] += q[3] + 0.5 \* d\_q[2] + (5 \* dd\_q[1] / 12) + (3 \* ddd\_q / 8);//расчет следующего y по формуле Адамаса

H[index] = start\_segment[3];

//сдвиг начального отрезка y0=y1 y1=y2 y2=y3

start\_segment[0] = start\_segment[1]; start\_segment[1] = start\_segment[2]; start\_segment[2] = start\_segment[3];

//сдвиг q, поиск q3

q[0] = q[1]; q[1] = q[2]; q[2] = q[3]; q[3] = h \* F(i + double\_h, start\_segment[3]);

//сдвиг delta q, delta^2 q, delta^3 q

d\_q[0] = q[1] - q[0]; d\_q[1] = q[2] - q[1]; d\_q[2] = q[3] - q[2];

dd\_q[0] = d\_q[1] - d\_q[0]; dd\_q[1] = d\_q[2] - d\_q[1];

ddd\_q = dd\_q[1] - dd\_q[0];

}

List<MyTable> result = new List<MyTable>(Convert.ToInt32(Math.Abs((xn - x0) / h) + 1));

Stopwatch timer = new Stopwatch();

double[] \_y = new double[Convert.ToInt32(Math.Abs((xn - x0) / h) + 1)]; //массив значений функции для оценки погрешности рассчета первых 4-х значений

result.Add(new MyTable(0, x0, y0, "0"));//добавляем первую строку в таблицу с известными значениями

double half\_h = h / 2;//половинный шаг для оценки погрешности

y = y0;

/\*рассчет 'y' с половинным шагом для оценки погрешности метода Рунге-Кутты\*/

for (var i = x0; i < x0+(2.5\*h); i += half\_h)

{

int index = Convert.ToInt32((i - x0) / half\_h);

double k1 = half\_h \* F(i, y);

double k2 = half\_h \* F(i + half\_h / 2, y + k1 / 2);

double k3 = half\_h \* F(i + half\_h / 2, y + k2 / 2);

double k4 = half\_h \* F(i + half\_h, y + k3);

y += (k1 + 2 \* k2 + 2 \* k3 + k4) / 6;

if (index % 2 == 0)

\_y[index/ 2] = y; //добавляем в массив каждое второе значение

}

y = y0;

timer.Start();

//поиск значений начального отрезка (методом рунге-кутты)

for (var i = x0; i < x0+(2.5\*h); i += h)

{

int index = Convert.ToInt32((i - x0) / h);

start\_segment[index] = y;//Значение y для началального отрезка

double k1 = h \* F(i, y);

double k2 = h \* F(i + h / 2, y + k1 / 2);

double k3 = h \* F(i + h / 2, y + k2 / 2);

double k4 = h \* F(i + h, y + k3);

y += (k1 + 2 \* k2 + 2 \* k3 + k4) / 6;

result.Add(new MyTable(index + 1, Math.Round(i + h, 5), Math.Round(y, 5), Convert.ToString(Math.Round(Math.Abs(y - \_y[index]) / 15, 5))));

}

start\_segment[3] = y;//значение y3 для начального отрезка

//расчет q0, q1, q2, q3

q = new double[] { h \* F(x0, start\_segment[0]), h \* F(x0+ h, start\_segment[1]), h \* F(x0 + (h \* 2), start\_segment[2]), h \* F(x0 + 3 \* h, start\_segment[3]) };

//расчет delta q0, delta q1, delta q2

d\_q = new double[] {q[1] - q[0], q[2] - q[1], q[3] - q[2]};

//расчет delta^2 q0, delta^2 q1

dd\_q = new double[] {d\_q[1] - d\_q[0], d\_q[2] - d\_q[1]};

//расчет delta^3 q0

ddd\_q = dd\_q[1]-dd\_q[0];

for (var i = x0+(h\*3); i < xn-half\_h; i += h)

{

int index = Convert.ToInt32((i +h)/ h);

start\_segment[3] += q[3] + 0.5 \* d\_q[2] + (5 \* dd\_q[1] / 12) + (3 \* ddd\_q / 8);//расчет следующего y по формуле Адамаса

if (index > 7 && index % 2 == 0)

{

result.Add(new MyTable(index, Math.Round(i + h, 5), Math.Round(start\_segment[3], 5), Convert.ToString(Math.Round(Math.Abs(H[index%8]-start\_segment[3])/15, 5))));//добавление результатов в список

}

else

{

result.Add(new MyTable(index, Math.Round(i + h, 5), Math.Round(start\_segment[3], 5), ""));//добавление результатов в список

}

//сдвиг начального отрезка y0=y1 y1=y2 y2=y3

start\_segment[0] = start\_segment[1]; start\_segment[ 1] = start\_segment[2]; start\_segment[2] = start\_segment[3];

//сдвиг q, поиск q3

q[0] = q[1]; q[1]= q[2]; q[2] = q[3]; q[3] = h \* F(i + h, start\_segment[3]);

//сдвиг delta q, delta^2 q, delta^3 q

d\_q[0] = q[1] - q[0]; d\_q[1] = q[2] - q[1]; d\_q[2] = q[3] - q[2];

dd\_q[0] = d\_q[1] - d\_q[0]; dd\_q[1] = d\_q[2] - d\_q[1];

ddd\_q = dd\_q[1] - dd\_q[0];

}

timer.Stop();

label.Content = timer.Elapsed.TotalMilliseconds + " мс";

grid.ItemsSource = result;

grid.Columns[0].Header = "#";

grid.Columns[1].Header = "X";

grid.Columns[2].Header = "Y";

grid.Columns[3].Header = "Погрешность";

}

private void Runge()

{

List < MyTable > result= new List<MyTable>(Convert.ToInt32(Math.Abs((xn - x0) / h)+1));

Stopwatch timer = new Stopwatch();

timer.Start();

double half\_h = h / 2;

double[] \_y = new double[Convert.ToInt32(Math.Abs((xn-x0)/h)+1)];

double y = y0;

result.Add(new MyTable(0, x0, y0, "0"));

for (var i = x0; i < xn-half\_h; i+=half\_h)

{

int index = Convert.ToInt32((i - x0 )/ half\_h);

double k1 = half\_h \* F(i, y);

double k2 = half\_h \* F(i + half\_h / 2, y + k1/2);

double k3 = half\_h \* F(i + half\_h / 2, y + k2/2);

double k4 = half\_h \* F(i + half\_h, y + k3);

y += (k1 + 2 \* k2 + 2 \* k3 + k4)/ 6;

if (index % 2 == 0)

\_y[index/2] = y;

}

y = y0;

for (var i = x0; i < xn-half\_h; i+=h)

{

int index = Convert.ToInt32((i - x0) / h);

double k1 = h\*F(i, y);

double k2 = h\*F(i + h / 2, y + k1/ 2);

double k3 = h \* F(i + h / 2, y + k2 / 2);

double k4 = h \* F(i + h, y + k3);

y += (k1 + 2 \* k2 + 2 \* k3 + k4)/ 6;

result.Add(new MyTable(index+1, Math.Round(i+h,5), Math.Round(y,5), Convert.ToString(Math.Round(Math.Abs(y - \_y[index]) / 15, 5))));

}

timer.Stop();

grid.ItemsSource = result;

grid.Columns[0].Header = "#";

grid.Columns[1].Header = "X";

grid.Columns[2].Header = "Y";

grid.Columns[3].Header = "Погрешность";

label.Content = timer.Elapsed.TotalMilliseconds + " мс";

}

private double F(double x, double y) => 2\*Math.Pow(x, 2)+Math.Pow(Math.E, y)+y;

private void checkBox1\_Checked(object sender, RoutedEventArgs e)

{

checkBox2.IsChecked = false;

}

private void checkBox2\_Checked(object sender, RoutedEventArgs e)

{

checkBox1.IsChecked = false;

}

}

}